

## ***AsAudit – User's Manual***

*Doc. No. ENP5020  
Version: 27-09-2007*

**ASKOM®** and **asix®** are registered trademarks of ASKOM Spółka z o.o., Gliwice. Other brand names, trademarks, and registered trademarks are the property of their respective holders.

All rights reserved including the right of reproduction in whole or in part in any form. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without prior written permission from the ASKOM.

ASKOM sp. z o. o. shall not be liable for any damages arising out of the use of information included in the publication content.

Copyright © 2007, ASKOM Sp. z o. o., Gliwice



ASKOM Sp. z o. o., ul. Józefa Sowińskiego 13, 44-121 Gliwice,  
tel. +48 (0) 32 3018100, fax +48 (0) 32 3018101,  
<http://www.askom.com.pl>, e-mail: [office@askom.com.pl](mailto:office@askom.com.pl)

## TABLE OF CONTENTS

1. PREFACE .....	3
1.1. DATABASE SERVER .....	3
1.2. COMPONENTS .....	3
1.3. CONFIGURATION DATA STRUCTURE .....	4
1.4. MULTI-STATION CONFIGURATION .....	4
2. FUNCTIONS OF ASAUDIT MODULE .....	5
2.1. USER LOG-IN AND AUTHORIZATION CONTROL SYSTEM .....	5
2.1.1. <i>List of protected system functions</i> .....	6
2.2. OPERATOR'S NOTEPAD .....	7
2.3. LOGGING THE PERFORMED CONTROL OPERATIONS .....	7
2.4. LOGGING THE OPERATOR'S ACTIONS .....	7
2.5. APPLICATION INTEGRITY CONTROL .....	8
3. ASAUDIT MODULE PROGRAMS .....	9
3.1. CONSOLE .....	9
3.2. BROWSER .....	10
3.3. CONFIGURATOR .....	12
3.3.1. <i>Creating a new application</i> .....	13
3.3.2. <i>User login</i> .....	15
3.3.3. <i>Basic parameters panel</i> .....	16
3.3.4. <i>Default user</i> .....	17
3.3.5. <i>Meaning of displayed icons</i> .....	18
3.3.6. <i>Updating the application integrity data</i> .....	19
4. MODIFICATION OF DATABASE STRUCTURE FOR CONTROL LOGGING PARAMETERIZATION .....	21
5. CONTROLLING ASWORK MODULE PROGRAMS FROM ASIX APPLICATIONS .....	23



# 1. Preface

AsAudit module is a part of Asix system. Its basic task is the adaptation of Asix to the requirements set during the validation of systems designed for food processing and pharmaceutical industry. AsAudit replaces the independent authorization control systems utilized by individual Asix modules with one central user signup system.

AsAudit module supports the following functions:

- User log-in and authorization control system
- Operator's notepad
- Logging control action performed for selected variables
- Logging the operator's actions
- Application integrity control

## 1.1. Database server

AsAudit operation is based on application of SQL database for storing the configuration data and logging data collected during application operation.

To use AsAudit, Microsoft SQL Server 2005 should be pre-installed. Free Express Edition, downloadable from page: [www.microsoft.com/downloads](http://www.microsoft.com/downloads) can be used.

During installation, it is recommended to select a mixed user authorization mode (SQL Server/Windows NT).

## 1.2. Components

AsAudit module consists of the following programs:

- **AsAudit console**

The basic program, which should be running for the entire duration of application operation. It is responsible for user log-in, checking the authorizations and collecting information on application operation.

To run any application utilizing the AsAudit function, the Console must be started as the first Asix application program.

- **AsAudit browser**

Program used to browse and analyze data collected during the application operation.

- **AsAudit configurator**

Interactive configuration program for application parameters.

## 1.3. Configuration data structure

Configuration data of AsAudit application are stored in two locations:

- **XML file**

XML file stores a minor part of the configuration data. These are: database access parameters and Asix application launch parameters.

The name of XML file is given in command lines launching the AsAudit programs. Due to file integrity protection, XML file cannot be created manually – it must be saved with use of AsAudit's configuration program.

- **Application database**

It stores all remaining configuration data. Similarly to XML file, due to integrity protection, the contents of AsAudit database configuration data cannot be modified with tools other than AsAudit Configurator.

## 1.4. Multi-station configuration

In case of network (multi-station) configurations, a single database located on a selected station is used. XML files can be different, on condition that identical database access parameters are defined for all of them.

AsAudit console has an embedded protection against loss of database server connection. During start-up the Console, if necessary, uses a local copy of configuration data. Logged data can also be buffered locally, until the connection with the database server is reestablished.

## 2. Functions of AsAudit module

### 2.1. User log-in and authorization control system

AsAudit module allows for controlling access to the following asix application components:

- **System functions**

Some system functions (e.g. exiting from the application, alarm exclusion) can be blocked for unauthorized users. Parameterization of access rights to the protected actions is performed through indicating users, who ARE ALLOWED to execute such action. By default, the action is blocked for all users.

- **Files**

Selected files can be protected against use by an unauthorized user. In such case, the operation is blocked.

By default, files are not protected.

In some cases, the application software may ignore the access rights. This pertains to files loaded by Asix in the context of all users, e.g. alarms system configuration files.

- **Process variable control operations**

As Audit can control the attempts of sending controls to selected variables. Visualization objects of Asix masks will automatically block the function selecting new values, if the user does not possess the appropriate rights.

By default, the control operations for a variable are not protected. At the same time, regardless of AsAudit protection, Asmen utilizes its own standard protection mechanisms on the level of communication channels.

AsAudit allows for creation of user groups. Groups can be used during assigning authorizations for protected application elements, in the same manner as normal users.

Furthermore, it is possible to define a default user, who will be automatically logging on during system start-up and after each logging out of another user. This mechanism allows to define a minimal authorization level available for all application users.

The database registers all user log-in operations, and also all events of attempted unauthorized access to application elements.

### 2.1.1. List of protected system functions

The following table presents a description of system function controlled by AsAudit.

DESCRIPTION	Group
Right to parameterize AsAudit application. Allows for use of AsAudit Configurator.	AsAudit
Right to close AsAudit console, which is equivalent to closing the entire application.	AsAudit
Right to browse the archive of performed controls in AsAudit Browser.	AsAudit
Right to browse the operator's actions archive in AsAudit Browser.	AsAudit
Right to browse the archive of notes and events in AsAudit Browser.	AsAudit
Right controlling the desktop protection function. Lack of permission means that the system desktop is blocked for the user.	System
Right of As program administrator. Functionally equivalent to administrator level in As log-in system.	As
Rights of the 1 <sup>st</sup> level password of As program. Functionally equivalent to rights of the 1 <sup>st</sup> level password in log-in system or password system of As program.	As
Rights of the 2 <sup>nd</sup> level password of As program. Functionally equivalent to rights of the 2 <sup>nd</sup> level password in log-in system or password system of As program.	As
Rights of the 3 <sup>rd</sup> level password of As program. Functionally equivalent to rights of the 3 <sup>rd</sup> level password in log-in system or password system of As program.	As
Rights of the 4 <sup>th</sup> level password of As program. Functionally equivalent to rights of the 4 <sup>th</sup> level password in log-in system or password system of As program.	As
Rights to switch to Designer mode of AS program. Identical as <i>Designer lock</i> option in the initiating file of AS program.	As
Right to change the time of AS program. Identical as <i>Time change</i> limit=0 in the initiating file of AS program.	As
Right to edit AS program reports. Identical as <i>Editing lock</i> option in the initiating file of AS program.	As
Right of interactive choice of custom mask. In AS program, the function is protected within <i>Change lock</i> option in the initiating file.	As
Right to use the AS file manager. Identical as FILE_MANAGER_LOK option in the initiating file of AS program.	As
Right to close As application. Identical as <i>Application closing lock</i> option in the initiating file of AS program.	As
Right to change the alarm filtering settings. Identical as option setting the authorization level for changing the filtered alarms – an option in the initiating file of AS program.	As
Right to change the alarm sound signal settings. Identical as option setting the authorization level for changing the alarms signaled with sound – an option in the initiating file of AS program.	As
Right to change the alarm exclusion settings. Identical as option setting the authorization level for changing the excluded alarms – an option in the initiating file of AS program.	As
Right to edit As program tables.	As
Right of AsTrend program administrator	AsTrend
Right to save the trends' configuration file	AsTrend



## 2.2. Operator's notepad

Operator's notepad allows the system users to enter custom texts. Notes are entered in the AsAudit Console window. Operator's notepad is not parameterized.

## 2.3. Logging the performed control operations

AsAudit can register the control operation performed on selected process variables.

The following data are registered:

- Moment of control execution
- ID of the logged-in user
- Name of machine performing the control
- Value of process variable before the control operation
- Control value
- Value sending status

Parameterization (choice of variables) of the control operation logging is executed in the variables database by setting the *ControlLogging* attribute to a non-zero value.

## 2.4. Logging the operator's actions

AsAudit can register some actions of Asix system operators. It is possible to log the following operations:

- Mask opening
- Mask closing
- Table opening
- Table closing
- Trend opening
- Trend closing
- Change of AS program status (beginning, end, switching to Designer)

The type and object of operation (e.g. mask name) is logged, together with event time and the ID of the current user.

Parameterization of the operator's actions logging is executed in the configuration program and consists in definition of the types of registered operations and names of computers, for which the logging is performed.

## 2.5. Application integrity control

Application integrity control consists in verification of the contents of the database and the selected files used in the application. Configuration program calculates the checksums of the indicated files and the database. Checksums are verified during the system operation. In the start phase of the Console, all controlled files are checked. If discrepancies are found, user may decide whether the application start should be continued. Additional tests are performed on each attempt of access to the controlled file – in case of error, the operation is blocked. Each integrity error is logged in the database.

## 3. AsAudit module programs

### 3.1. Console

AsAudit console is started by means of the following command:

```
AsAuditConsole configuration_file [/protect]
```

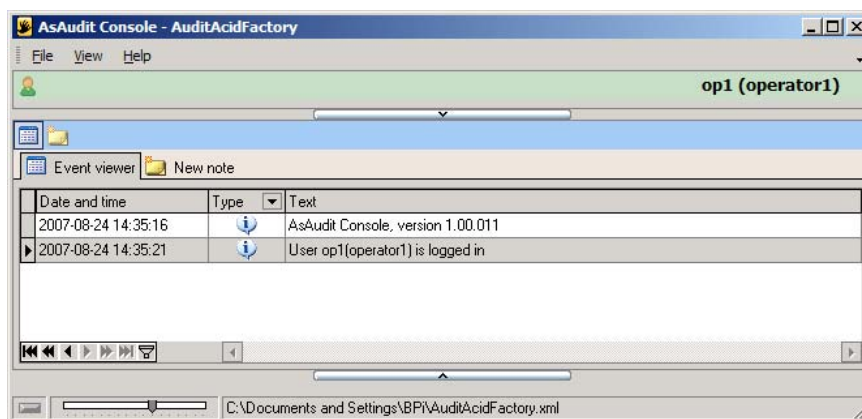
The mandatory parameter *configuration\_file* determines the name of xml configuration file of AsAudit application.

Optional switch /protect enables the desktop protection support (locking the taskbar, Windows function keys, etc.) By default, the protection is disabled. If Console is used, As32 program of Asix package automatically disables its own desktop control.

A very important element of Console start is the application's start directory, which has to be consistent with the base directory of the files being the components of the application. This is a result of the fact that all filenames used in configuration of AsAudit application (protection of file access and integrity control) are saved in the database with relative paths (relative towards the base directory).

Console start can be linked with starting the As32 program (if the configuration file contains the As32 start parameters). This allows to start the entire Asix application from a single shortcut on the desktop starting the AsAudit Console.

The Console monitors the changes of configuration data. If change is detected, the Console automatically reads new configuration data from databases.



Console window consists of two switched panels. The upper one is used for user log-in. The lower panel contains two tabs:

- **Event viewer**

It is used to view messages related with current operation of AsAudit application. These are not the messages logged in the database. Part of the messages are copies of events saved in the database, the others are related to operation of AsAudit module software. Messages from "Event viewer" tab are saved to disk in *aswork\_console.log* file.

- **New note**

This tab allows the operator to enter a text note.

## 3.2. Browser

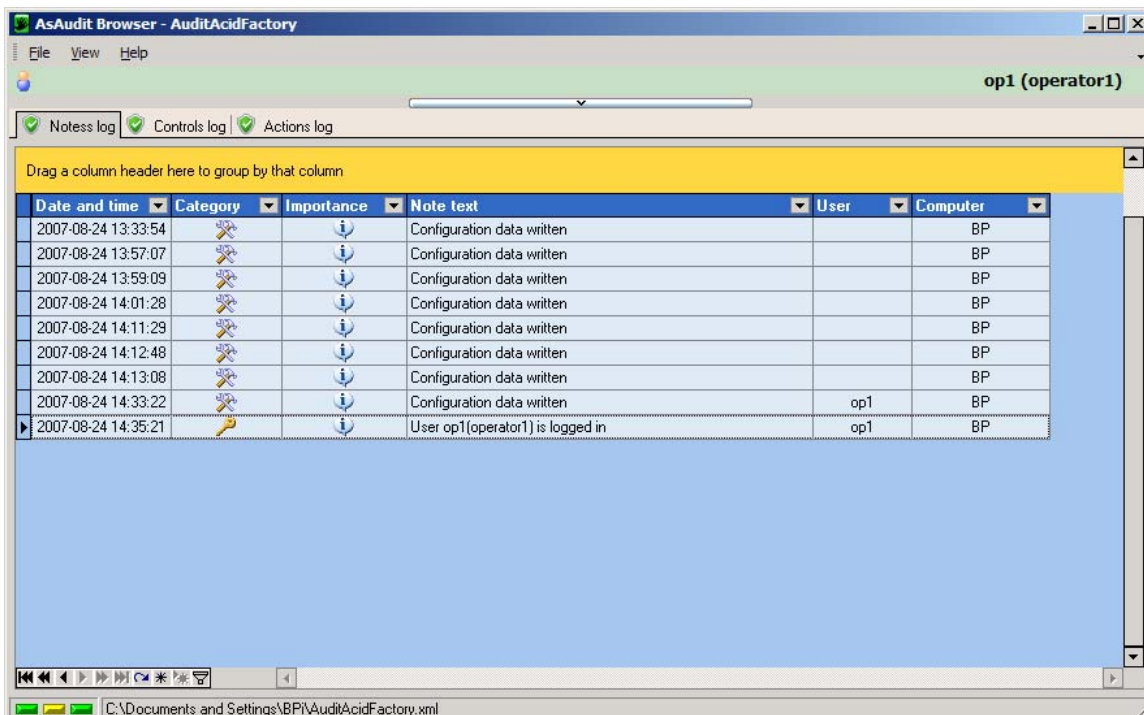
AsAudit browser is started by means of the following command:

```
AsAuditBrowser [configuration_file]
```

The parameter *configuration\_file* determines the name of xml configuration file of AsAudit application.

If the filename is not given, it is possible to select the configuration file later on, by means of functions available in the Browser menu.

If Browser is started with operating Console, *configuration\_file* parameter is ignored – the configuration file given during Console start is used.



The window consists of two panels. The upper panel is used for user log-in. It is active only when the Console is not running.

The lower panel contains three tabs used for analysis of data registered in AsAudit application database:

- **Actions log**

Browsing logged operator's actions, e.g. mask opening, etc.

- **Controls log**

Browsing logged control operations related to process variables.

- **Notes log**

The collective event log. The following categories are used: logging events, access rights violations, integrity violations, operator's notes, configuration changes.

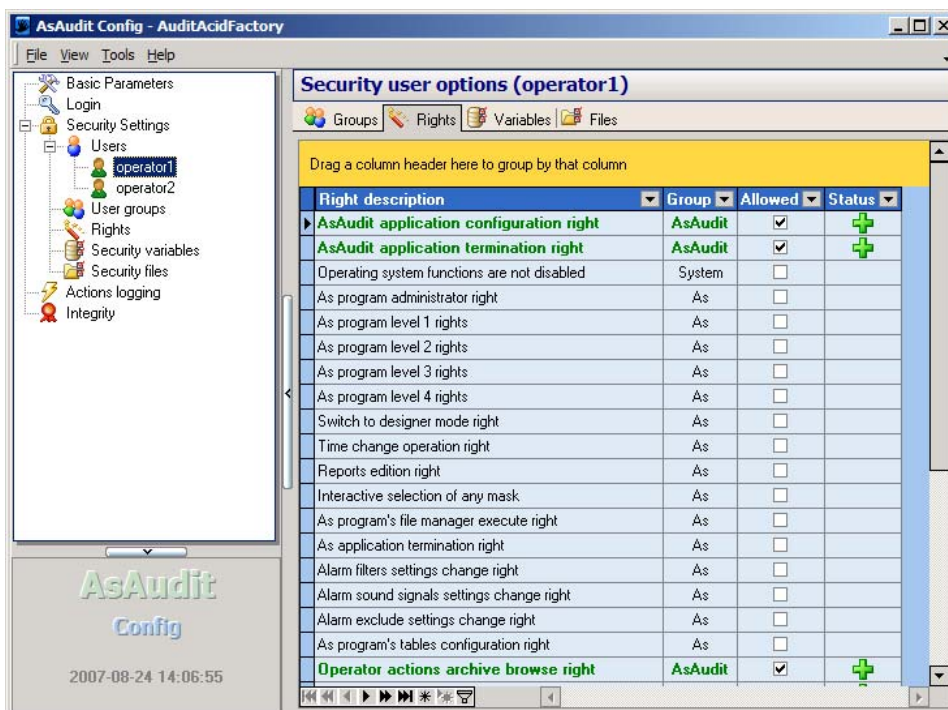
### 3.3. Configurator

AsAudit configurator is started by means of the following command:

```
AsAuditConfig [configuration_file]
```

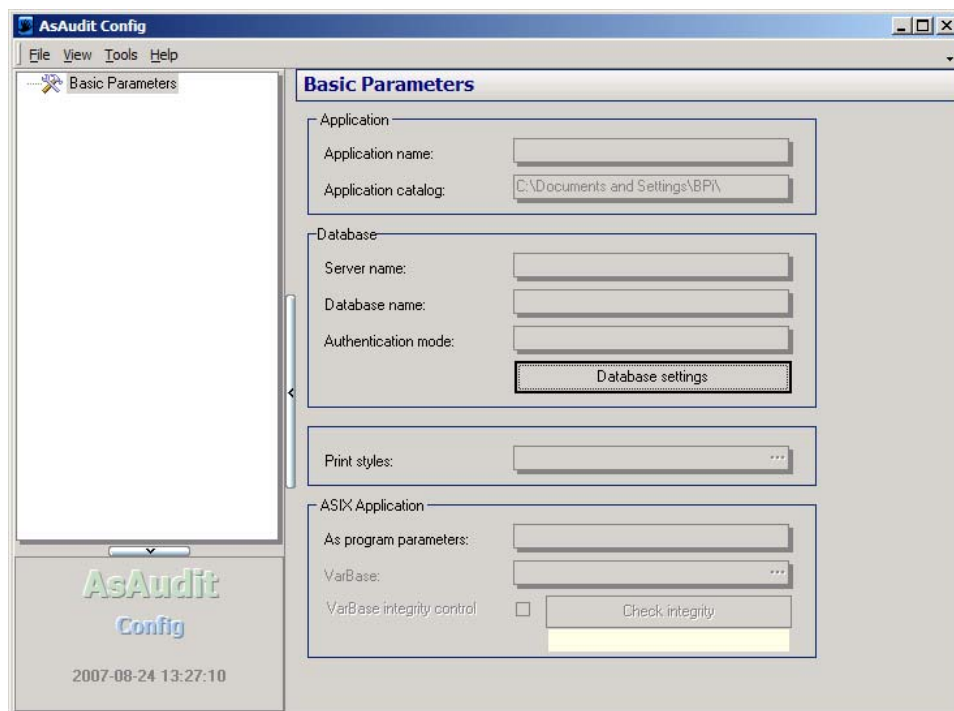
The parameter *configuration\_file* determines the name of xml configuration file of AsAudit application. If the filename is not given, it is possible to select the configuration file later on, by means of functions available in the Config menu.

If Config is started with operating Console, *configuration\_file* parameter is ignored – the configuration file given during Console start is used.

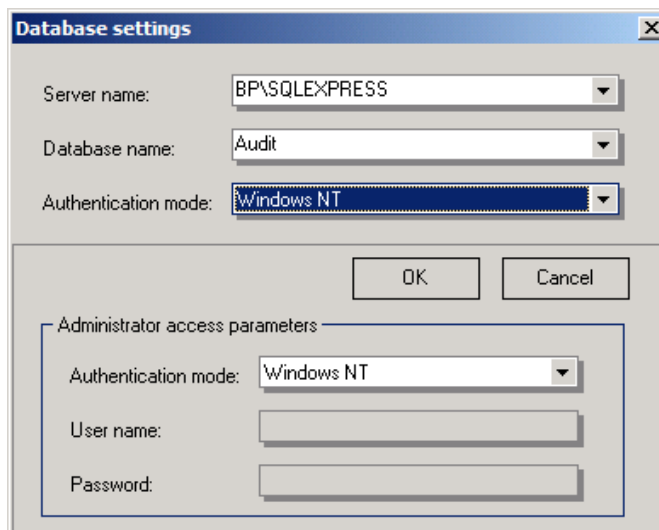


The Config window consists of two parts. In the left side of the window, a navigation tree is located. Its shape depends on the current Config context and a list of defined users and groups. Click on an appropriate tree element to switch the contents of the panel in the right part of the window, used for application parameterization.

### 3.3.1. Creating a new application



Use button *Database settings* to start the procedure of AsAudit database creation. The following dialogue box will be displayed:



Determine here to sever location and the database name. If *Server name* field is left empty, the database server running on the local machine will be used.

Authentication mode usually should be set in the original setting of *SQL Server*.

Administrator's server access parameters are used only in one moment – during the database creation. If Windows account, in which the work is

performed, is an administration account, *Windows NT* mode can be used. If not, *SQL Server* authentication should be usually used, and the server administrator's account and password should be openly given.

However, the authentication rules may be different, depending on the security policies adopted in the company.

When *OK* key is pressed, the following window will be displayed (unless the database of the given name already exists):



*Database backup file* field is used only, when the new database should be created on the basis of possessed backup copy of another AsAudit database (e.g. during database migration from another server).

*Database folder* field is used to determine location of files, where SQL Server saves the database (only if the files should be located in a non-standard location).

When the database is created, the next step is to define a user with rights to edit the AsAudit application. For this purpose:

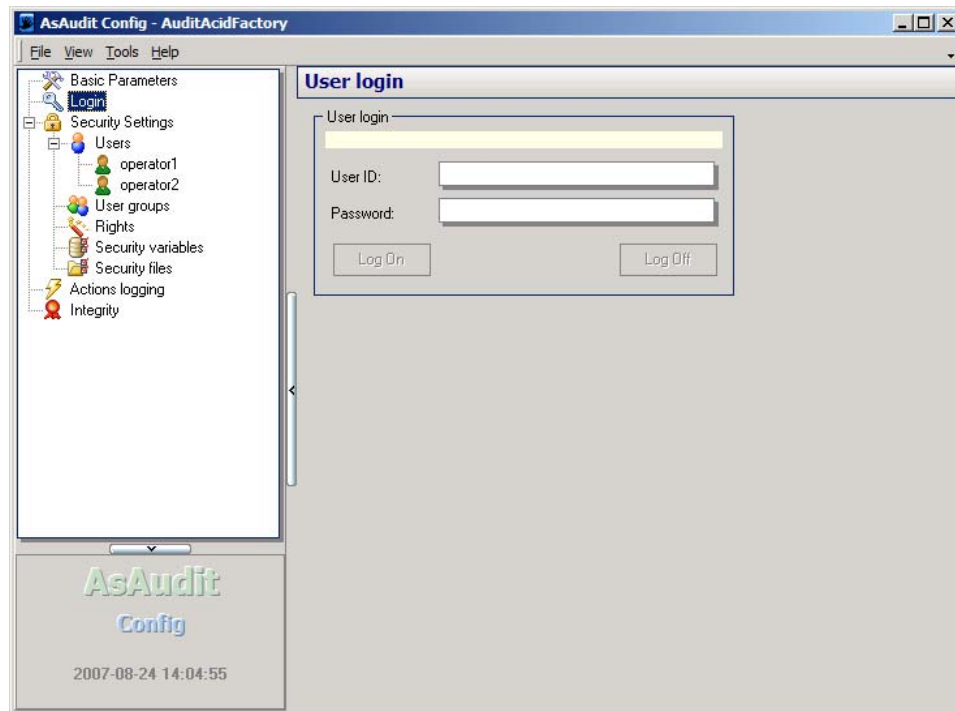
- Select *Users* node in the navigation tree
- Enter new user in the working panel
- Select *Rights* node in the navigation tree
- In the upper part of the working panel, select *AsAudit application parameterization right*, and then in the lower part of the panel check the checkbox in the line of previously created user.

The last operation in the new application creation process is the *Save* function from *File* menu in order to save the configuration data in the xml file and the database. The xml file created in such manner is later used as the start parameter for Console and other AsAudit programs.



### 3.3.2. User login

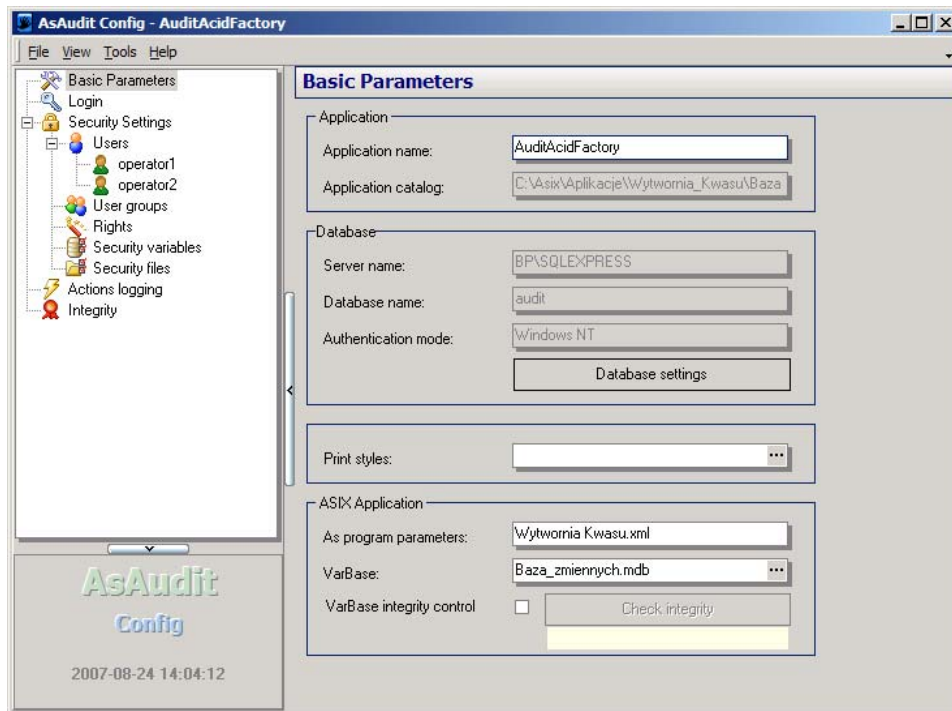
All changes of AsAudit application parameters require the right named *AsAudit application parameterization right*. When Config is started and xml file is loaded, the first operation is to log the user on. It is performed in the *Login* panel.



Config does not fetch the current user from the operating Console. A direct login in Config program must be always performed.

### 3.3.3. Basic parameters panel

*Basic parameters* panel is used to create AsAudit databases and to set parameters of global significance.



*Application name* field allows to name an application. The name is displayed on title bars of AsAudit programs.

Contents of *Application catalog* field cannot be changed. The application's base directory, resulting only from the location of the application's xml file, is displayed there. The relative paths to all application files are created in relation to the base directory.

Fields in *Database* group show current parameters of AsAudit application database. *Database settings* button allows to change those parameters.

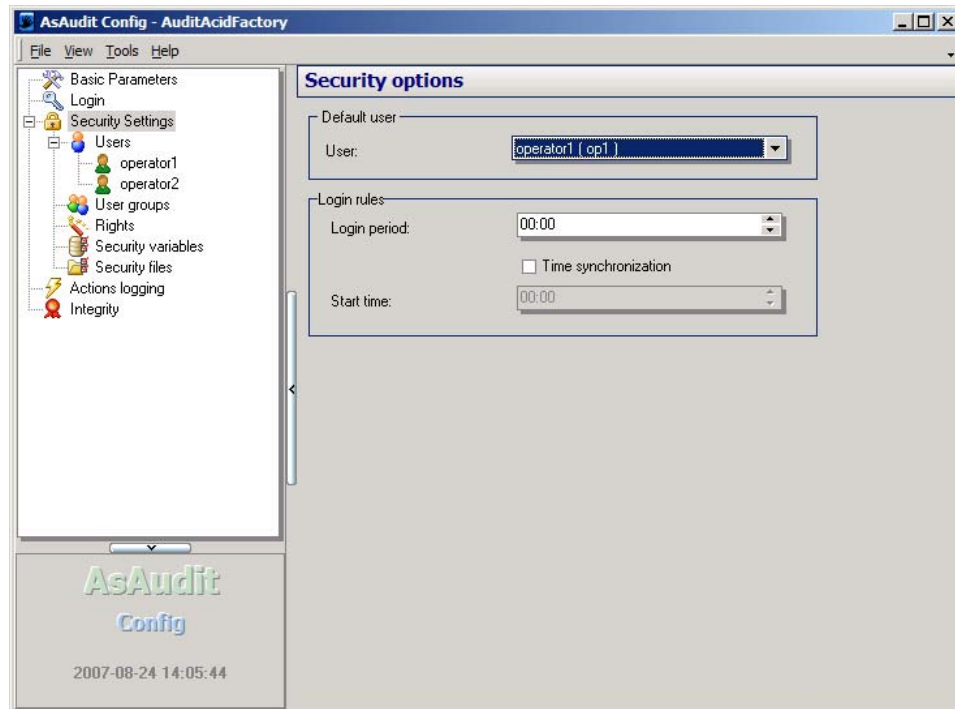
*Print styles* field, with an integrated button is used to create printout styles used in AsAudit application.

*AS program parameters* field should be set only if start of AsAudit Console should be combined with start of Asix visualization application. Those parameters are saved in XML application file.

*VarBase* field with integrated button is used to enter the variable definitions database parameters. The variables database contents are shown in the working panel during definition of variable writing protection policies. If *VarBase integrity control* field is checked, the contents of variable definitions database are also controlled in the aspect of application integrity. *Check integrity* button allows to check whether the variable definitions database was changed since the last calculation of the database contents checksum.

### 3.3.4. Default user

*Security Settings* node allows to define the so-called default user.



AsAudit programs during start-up automatically log the default user in. Similarly, when one of the users logs out, the default user is logged in in their place.

The default user mechanism allows to set a minimal scope of rights available for all application users.

### 3.3.5. Meaning of displayed icons

Icons displayed in the working panel tables have the following meaning:

*U/G* column



normal user  
group of users

*Status* column in *Rights* tab in the user/group panel



None  
none user possesses this right  
user has this right  
user does not have this right, but another user has it

*Status* column in *Variable* tab in the user/group panel



None  
variable writing is not controlled, all user can perform variable control operations  
user can perform variable writing  
user can not perform variable writing

*Status* column in *Files* tab in the user/group panel



None  
file access is not controlled, all users may use it  
file can be used by the user  
the user cannot use the file

*Used* column of *Rights* panel



None  
none user possesses this right  
the right was granted to at least one user

*Protected* column of *Variable protection* panel



None  
control operations for the variable are not protected  
control operations are protected

*Protected* column of *File protection* panel



None  
file access is not limited  
file access is protected

*Status* column of *Integrity* panel



None  
file is not controlled  
newly added file, not registered before  
file compliant with the previously registered version  
controlled file does not exist  
protected file has a changed content  
protected file has a changed time, but the contents are compliant  
error while checking the file version

### 3.3.6. Updating the application integrity data

If only updating of application integrity data (checksums of files) is necessary, open the application with *Open* command from *File* menu, and then use command *Save*. Config will ask whether the integrity-related data should be updated. If confirmed, Config will recalculate the checksums for all files subject to control.

Request to update the data integrity appears before each saving of the configuration data. Negative answer may result in change configuration data with maintenance of unchanged integrity data.



## 4. Modification of database structure for control logging parameterization

Choice of variables subject to control operation logging is executed in the variables database by setting the *ControlLogging* attribute to a non-zero value.

This attribute was not previously present in the database. Due to this fact, it is required to create a new variables database extended with *ControlLogging*. For this purpose, add the following entries to the database scheme file, which serves as the basis for database location:

- In section *[Attributes]* – add line "*ControlLogging = N,*"
- In section *[LocalAttributesNames]* – add line "*ControlLogging = 1045:*  
*Control logging, 1033: Control logging*"

When the scheme file is modified, use it as the basis for creation of new variables database by means of Variable Database Editor or Variable Database Manager.





## 5. Controlling AsWork module programs from Asix applications

It is possible to control the displaying of AsAudit Module program windows from the level of Asix visualization application. This is executed with ASAUDIT action of the following syntax:

*ASAUDIT window\_type,window\_parameters*

*Window\_type* parameter allows to select window to be activated. It can take the following values:

- *KONSOLA, CONSOLE* – AsAudit Console window
- *KONFIGURATOR, CONFIG* – AsAudit Config window
- *PRZEGLĄDARKA, BROWSER* – AsAudit Browser window

*Window\_parameters* parameter is not currently used.

If Config and Browser windows are activated, and those programs were not previously started, they will be started by ASAUDIT action.

The use of AsAudit Console also changes the functioning of other operator's actions related to password and user handling – the actions are locked or result in opening of the Console window.